Agile Data Offloading over Novel Fog Computing Infrastructure for CAVs

Andrea Tassi*, Ioannis Mavromatis*, Robert Piechocki*†, Andrew Nix*,
Christian Compton‡, Tracey Poole‡ and Wolfgang Schuster‡
*Department of Electric and Electronic Engineering, University of Bristol, UK
†The Alan Turing Institute, London, NW1 2DB, UK
‡Atkins Global Limited, London, UK

***Abstract*—Future Connected and Automated Vehicles (CAVs) will be supervised by cloud-based systems overseeing the overall security and orchestrating traffic flows. Such systems rely on data collected from CAVs across the whole city operational area. This paper develops a Fog Computing-based infrastructure for future Intelligent Transportation Systems (ITSs) enabling an agile and reliable off-load of CAV data. Since CAVs are expected to generate large quantities of data, it is not feasible to assume data off-loading to be completed while a CAV is in the proximity of a single Road-Side Unit (RSU). CAVs are expected to be in the range of an RSU only for a limited amount of time, necessitating data reconciliation across different RSUs, if traditional approaches to data off-load were to be used. To this end, this paper proposes an agile Fog Computing infrastructure, which interconnects all the RSUs so that the data reconciliation is solved efficiently as a by-product of deploying the Random Linear Network Coding (RLNC) technique. Our numerical results confirm the feasibility of our solution and show its effectiveness when operated in a large-scale urban testbed.**

***Index Terms*—Fog Computing, ITS, CAV, V2X, Network Coding, Data Offloading.**

I. Introduction

Connected and Autonomous Vehicles (CAVs) will play a pivotal role in our everyday lives in the future. CAVs, fully integrated with sensors, will be able to provide awareness of the surrounding environment and coordination with fixed network nodes and nearby vehicles [1]. To do so, they will require to exchange a vast amount of data, accommodated over various means of communication frameworks, and operated in a heterogeneous fashion [2], [3]. The broadcast and public nature of these communication links make them susceptible to cyber-security threats, and hence, impacting the passengers' safety. The vulnerability of automotive systems was recently demonstrated by the Chrysler Jeep hack, where the attackers exploited breaches in the vehicles internet-connected entertainment system[1].

Responding effectively to cyber-security incidents requires a number of different technological, procedural and organizational elements to be in place and work effectively. In many ways, the cyber-security requirements of CAVs are like any other business critical or safety-related system. Commercial Off-The-Shelf (COTS) equipment usually provides its information in forms of events and alarms. Correlation can be used to turn this data into more meaningful information. Machine Learning and Artificial Intelligence could be used to detect anomalies, which could affect the system [4]. New devices (such as roadside devices) could pass information (data and security oriented) to a centrally managed system [5], [6]. The authenticity and integrity of that information is of paramount importance. Being empowered by this knowledge and having the capability of detecting when the system is compromised, allow system administrators to determine the right course of actions [7].

In this paper, we focus on the pressing concern of empowering cloud-based services to detect incidents and respond to them by defining an agile strategy for data collection. In particular, we will propose a novel data offloading strategy allowing CAVs to share their sensory information with the Fog Computing Infrastructure and then, with cloud-based services. We envisage a system where our network will incorporate Fog Computing capabilities, giving us the leverage to use powerful processing nodes one-hope away from each Road-Side Unit (RSU). This will significantly reduce the latency of the data processing, minimizing the time required to identify potential threats.

The data offloading from a CAV to an RSU takes place when CAVs are in within range with one or more RSUs. The sensor data exchange happens in a broadcast fashion with the receivers and is organized as streams of information. However, the heterogeneity in driving behaviors within a city and the sparsity of the RSUs lead to intermittent communications. Reconciling the data at the infrastructure level becomes highly inefficient. Employing network coding techniques for achieving secure systems has attracted the interest of the research community lately, especially for wireless broadcast applications [8] and secure vehicular networks [9]. Inspired by that, and taking into account the fragmentary nature of the data offloading, we propose a new framework that uses Random Linear Network Coding (RLNC) to achieve an agile data offloading in Intelligent Transportation Systems (ITSs).

More specifically, our idea is based on the existing ITS-G5 Dedicated Short Range Communication (DSRC) protocol stack [10]. We propose an amendment as well as a RLNC version of the Cooperative Awareness Message (CAM), that is expected to be used for broadcasting sensor data. The amendment takes place at the Facilities layer. Later in this work, we will describe the proposed sublayer and its functionality. The

[1]https://www.bbc.co.uk/news/technology-33650491

coded packets generated from this sublayer are then mapped onto the proposed RLNC-CAM and broadcast to the network. When received from a RSU, the Fog Computing Infrastructure and the RLNC decoder are responsible for decoding the packet that could be then be forwarded to other cloud-based services.

The rest of the paper is organized as follows. Sec. II describes our system model. The proposed Fog Computing implementation and Cloud Computing capabilities are presented at first. Then, the need for RLNC and how it could operate within the context of a vehicular network are introduced. Following, we present the proposed extension to the ITS-G5 stack (Sec. III). More specifically, we start by describing the new RLNC-CAM message and the information encapsulated in that. In addition, we outline the design of the RLNC-Facility sublayer responsible for the generation of RLNC-CAMs. In Sec. IV, we present our numerical results that confirm the feasibility of our proposal. We focused our performance investigation on real-world data traces captured during an extensive experimental campaign that took place in the City of Bristol, UK. Finally, we summarize our findings in Sec. V and present a critical analysis of our results.

II. System Model

We consider a city-scale network providing wireless connectivity to CAVs. In our system model, each CAV wishes to offload its sensing acquired data onto a cloud-store facility by means of RSUs connected to a network of Fog computing infrastructure nodes, called Fog Orchestrators (FOs).

A. Fog Computing Infrastructure for Cyber-Secure ITSs

We assume our network being clustered in different management areas called Fog Areas. Each Fog Area consists of a number of RSUs, all being connected to a particular FO. We also assume that one FO serves each Fog Area. The FOs represents the logical entities encapsulating the core components of our system. FO ensures the programmability of the system simplifying the deployment of new functionalities, which rely upon a network of sensors and actuators.

In this particular system model, the FO is responsible for collecting the transmitted RLNC-CAMs, removing duplicated messages that may arise as a CAV is in range with two or more RSUs, recovering the encapsulated message, and passing them to cloud services. There, the threats can be effectively detected, and cyber-security decisions are taken.

We assume that a cloud-based data controller is responsible for the processing of all the sensor data transmitted by each CAV. Ultimately, RSUs are solely responsible for relaying messages to and from the CAVs. As shown in Fig. 1, this solution interacts with a cloud-based city-wide connection. In particular, the cloud-based service will only be in charge of recording city-scale data, interconnecting the different Fog Areas and enforcing city-scale policies to be put into practice.

This system model provides the necessary abstraction for a next-generation ITS system. This will also give us the leverage to deliver cyber-secure ITSs. In fact, as CAV technologies come to play an ever-increasing role in the safe and smooth running of road networks, network operators will increasingly need to implement effective cyber-security measures. These measures need to ensure that road users personal data is not compromised but just as importantly (arguably more importantly) they must ensure that the systems which CAVs rely on work as designed. An attacker could take more focused actions, which could result in an even greater impact on journeys, they may choose to spoof communications to systems for example which could then possibly have an impact on safety.

B. RLNC for Agile Data Offloading

For the data offloading to happen, each CAV broadcasts its sensor data as it gets within range with one or more RSUs. However, as the CAV leaves the coverage area of a RSU, the data offloading process may be interrupted to be resumed as the CAV reaches the next coverage region. This particularly applies, in sparsely deployed RSU networks. For this reason, traditional handover procedures cannot be put in place – thus making the reconciliation of data received by multiple RSUs very inefficient and not scalable, as the number of CAVs increases. With these regards, RLNC promises to overcome these issues in a seamless fashion [11].

Consider the case of a CAV wishing to transmit to one or multiple RSUs a source message over a broadcast channel. In our system, a source message represents a stream of sensor information packets. Transmissions experience a certain packet error probability. According to the RLNC principle, the CAV segments the source message into K source packets $\{\mathbf{s}_i\}_{i=1}^{K}$ where $\mathbf{s}_i$ is made of L elements from a finite field $\mathbb{F}_q$. The CAV also linearly combines at random the source packets to obtain N coded packets $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{N}$ for transmission. Each coded packet $\mathbf{c}_j$ consists of L elements from $\mathbb{F}_q$ and it is obtained as:

$$\mathbf{c}_j = \sum_{i=1}^{K} g_{i,j} \cdot \mathbf{s}_i, \tag{1}$$

where the coding coefficient $g_{i,j} \in \mathbb{F}_q$. Besides, let $\mathbf{G}$ be a $K \times N$ random matrix over $\mathbb{F}_q$ where its (i,j)-th element is defined by $g_{i,j}$, the relation

$$[\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N] = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{c}_K] \cdot \mathbf{G} \tag{2}$$

holds true.

Let $\mathcal{C}$ be the set of n coded packets pertaining to the same CAV and successfully received by each of the RSU, for $0 \geq n \geq N$. In turn, each RSU forwards to the FO and then to a cloud-based service the received coded packets. Hence, the cloud-based service can populate a $K \times n$ decoding matrix where the n columns of $\mathbf{M}$ are defined by the n columns in $\mathbf{G}$ associated with coded packets in $\mathcal{C}$. The source message is recovered as soon as the rank of $\mathbf{M}$ becomes equal to K. In particular, the probability $\mathrm{R}(n)$ of a source message of being recovered, as a function of n, can be expressed follows [12]:

$$\mathrm{R}(n) = \prod_{t=0}^{K-1} \left(1 - \frac{1}{q^{n-t}}\right) \tag{3}$$

In our system, the transmission of each source message takes place in an unacknowledged fashion. After that the N-th coded packet associated with a source message has been

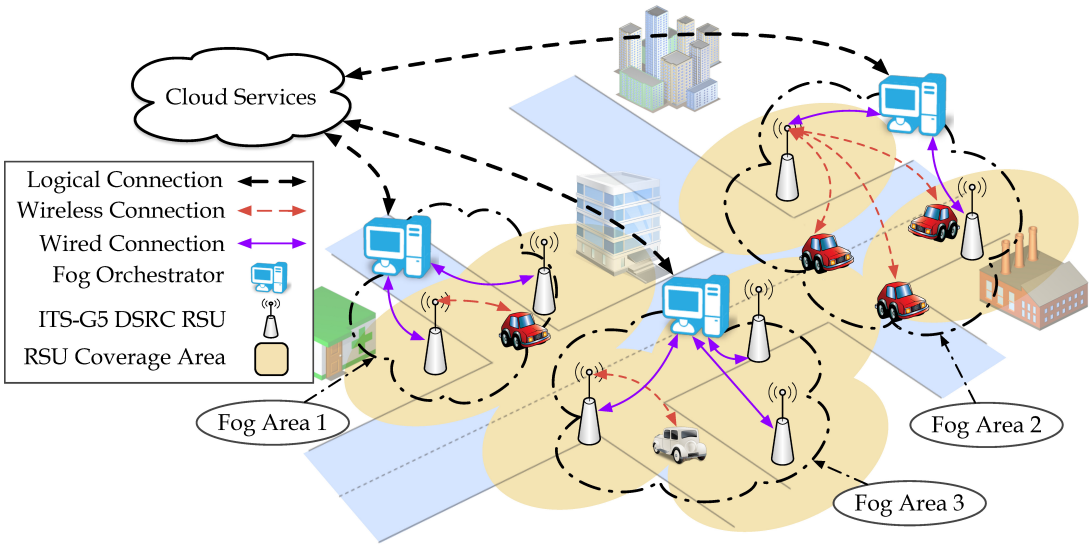

Fig. 1. General overview of the considered Fog Computing infrastructure for CAVs.

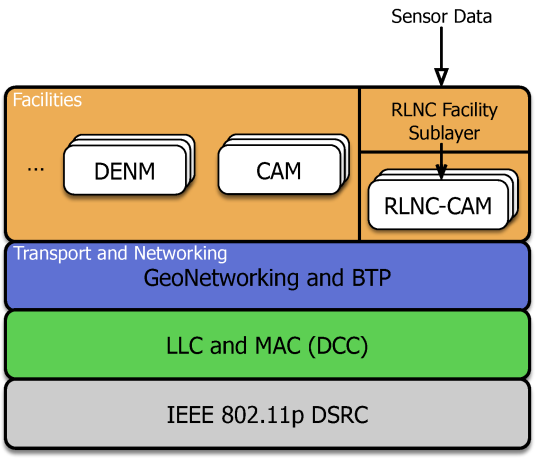

Fig. 2. Proposed amendment to the ETSI's ITS-G5 protocol stack including the RLNC-Facility sublayer.

transmitted by a CAV, the broadcasting of the following source packet begins. In the following section, we will describe how a sensor data stream can be partitioned into a sequence of source message and how the RLNC principle can be integrated into the ETSI's ITS-G5 communication stack.

III. Proposed Extension to the ITS-G5 Stack

In our system, we assume that CAVs and RSUs communicate using the ETSI's ITS-G5 standard [10]. The ITS-G5 standard has been derived from the US Wireless Access in Vehicular Environment (WAVE), and both of them build upon the IEEE 802.11p DSRC physical layer. As shown in Fig. 2, the Medium Access Control (MAC) layer of the ITS-G5 protocol stack employs the Decentralized Congestion Control (DCC) protocol to dynamically optimize the channel load. This is done by adapting the transmission power, the rate and the sensitivity of a transceiver. Then a simplified Logical Link Control (LLC) layer acts as an adaptation layer between the DCC and the upper layers. With regards to the Networking and Transport Layer, packets can be forwarded in a multi-hop fashion by the GeoNetworking protocol, which employs the Contention Based Forwarding (CBF) algorithm. In particular, on the basis of geolocation information, the CBF algorithm elects as multi-point relay node the CAV at the greatest distance from the source node. On the other hand, point-to-point communications are handled by the Basic Transport Protocol (BTP), which is responsible for multiplexing/demultiplexing packets originated by the Facility layer [13].

At the level of the Facility layer, CAMs and Decentralized Environmental Messages (DENMs) are transmitted/received. In particular, CAMs are periodically broadcast by each CAV and convey information pertaining to the position of a CAV, its engine status, etc. On the other hand, DENMs are transmitted in the response of an event and are used to alert road users for a danger ahead, adverse weather conditions, etc [13]. As per Sec. II, in order to enable cloud-based services to detect cyber-security threats efficiently, CAVs are expected to share their sensor information periodically. As such, building upon the standard definition of a CAM, we propose the RLNC-CAM version that each CAV is expected to use to broadcast sensor information. In particular, as shown in Fig. 2, when a CAV wishes to employ RLNC-CAMs, sensor information is provided as an input to the RLNC-Facility sublayer, which is responsible for the following:

- Segmenting the sensor data stream into a sequence of source packets with the same bit length.
- Organizing the sequence of source packets into source messages defined by K consecutive source packets.
- Assigning an ID to each source message.
- Generating a sequence of N coded packets per each source message according to the RLNC principle (see Sec. II-B).

Each coded packet generated by the RLNC-Facility Sublayer is mapped into an RLNC-CAM having the structure

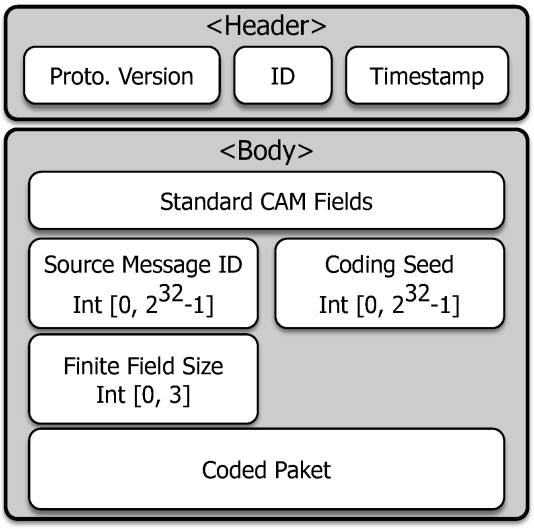

Fig. 3. Structure of the proposed RLNC-CAM message. Both the standard CAM fields and the RLNC fields are encapsulated in the body of the message.

shown in Fig. 3. Each RLNC-CAM comprises the same fields as a standard CAM [14]. In addition, the RLNC fields are added to the message. More specifically, the header of an RLNC-CAM includes the protocol version in use, a general CAM ID and a generation timestamp. The body hosts information pertaining the CAV transmitting the message, such as: a transmitter ID, the nature of the CAV (mobile, public authority, private, etc.), its position (latitude, longitude, elevation, and heading) and a set of optional attributes.

The RLNC fields that follow the optional CAM attributes are defined as follows:

- *Source Message ID* contains the ID of the source message that is being transmitted. Similarly to the Station ID filed, this field is 32 bits long, and it is defined as an integer value ranging between 0 and $2^{32}-1$.
- *Finite Field Size* encodes the value of q that is being used by the RLNC-Facility Sublayer to generate each coded packet. In an embedded application, finite fields with a characteristic equal to 2 are generally preferred because of the high level of optimization that it is possible to achieve in the implementation of the RLNC encoder/decoder [15]. Thus, for practical application, values of q are restricted to $2, 2^2, 2^3, \ldots$. We propose that Finite Field Size field represents the value $\log_2(q) - 1$. Thus, if we limit the value of q to 2^8, it is sufficient for the field to be 3 bits long to encode the values $2, 2^2, \ldots, 2^8$.
- *Coding Seed*: Contains the seed used to initialize the pseudo-random number generator (PRNG) used to generate the coding vector associated with a coded packet. As such, provided the receiving end is equipped with the same PRNG, each coding vector can be precisely recovered. This solution is more efficient than including into each RLNC-CAM the entire coding vector represented as an array of integer values. Only a single integer representing the seed is included. By following the same reasoning as in [16], we suggest the Coding Seed field to be 32 bits long – thus capable of expressing integer values ranging between 0 and $2^{32}-1$.
- *Coded Packet*: Contains the actual coded packet generated by the RLNC-Facility sublayer. Its bit length is variable and is specified by the RLNC-Facility sublayer.

As soon as an RSU receives an RLNC-CAM, it is forwarded to the FO and then to the cloud. Then, for any CAV and any source message with a given ID, a could-based service is responsible for checking if a number of RLNC-CAM carrying linearly independent coded packets have been received. If so, then an that particular source message can be recovered and forwarded to the other could-based services.

TABLE I
SIMULATION AND EXPERIMENTAL PARAMETERS.

Parameter	Value	
Experiment/Simulation Time	8	h
Carrier Frequency	5.9	GHz
Bandwidth	10	MHz
RLNC-CAM Length	2048	B
RLNC-CAM Interval	10	ms
Transmission Power	29	dBm
Background Noise	$\mathcal{N}(-110, 3)$	dBm
Connector and Cable Losses	3	dBm

IV. PERFORMANCE EVALUATION

We consider an infrastructure network composed of four RSUs, deployed in the City of Bristol, UK. More specifically, four RSUs were deployed (as shown in Fig. 4). *RSU1* was mounted close to a blind T-junction, on a curvy road at the height of $8\,\mathrm{m}$. The position of *RSU2* was a straight road with light foliage, being mounted on the wall of a building at $5\,\mathrm{m}$. *RSU3* was the highest placed RSU, mounted at the balcony at $25\,\mathrm{m}$, and providing coverage to a wide straight road. Finally, *RSU4* can be located at one of the main arteries of the City of Bristol, being mounted at $12\,\mathrm{m}$. The different positions and buildings were chosen to investigate their effect in the performance of each individual link. Each RSU has a direct link to the Fog Computing Infrastructure described above. These deployed RSUs are being used for the two phases of our field-trials, with the first one being described in [17].

During our experimental campaign, four vehicles were driven around the City of Bristol, UK for several hours, following the route shown in Fig. 4 (two in clockwise direction and two in anti-clockwise direction). All the transceivers operated at the frequency band of $5.9\,\mathrm{GHz}$ and broadcast a RLNC-CAM per $10\,\mathrm{ms}$. Each RLNC-CAM encapsulated the information shown in Fig. 3. By means of a logging interface, we logged all the transmitted and received RLNC-CAMs. More information about the deployed testbed, the devices we utilized and the implemented communication stack, can be found in [17]. All the simulation and experimental parameters can be found in Table I.

Starting with Fig. 4, we present the heatmap for the packet delivery probability for all RLNC-CAMs transmitted from a vehicle and received by an RSU. Each square on this heatmap is $4\,\mathrm{m}^2$. We observe that, as expected, the packet delivery rate increases as we approach an RSU, it peaks when a vehicle passes by and fends off until being outside of coverage area.

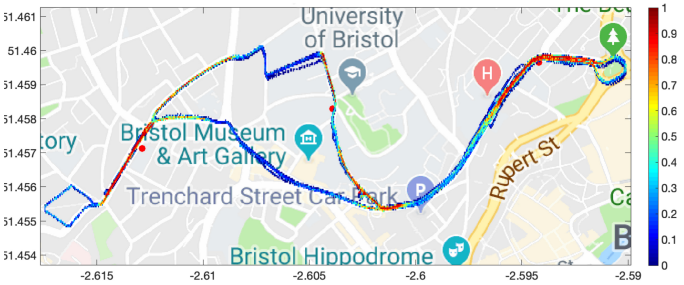

Fig. 4. The heatmap results for all RSUs in our system. This figure shows the packet delivery rate for all RLNC-CAMs transmitted from a vehicle to a RSU and all the positions of the RSUs.

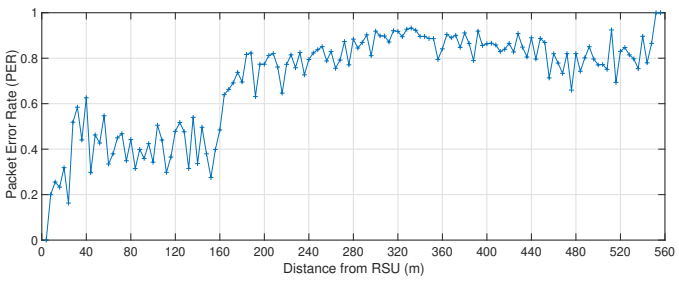

Fig. 5. The packet error probability for *RSU1* as a function of the distance. Each bar represents probability of a RLNC-CAM to be lost when a vehicle is within this particular $4\,\mathrm{m}^2$ area and the given distance from the RSU.

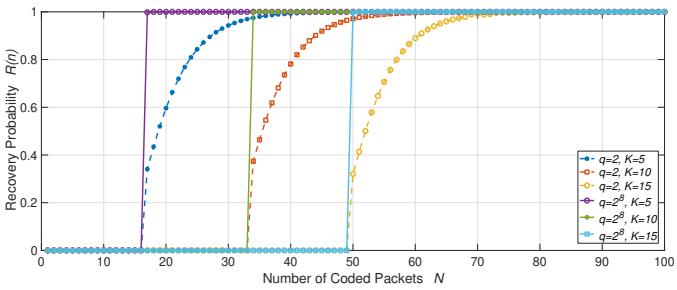

Fig. 6. The recovery probability R as a function of the number of RLNC-CAM transmissions, for $K = 5, 10, 15$ and $q = 2, 2^8$ as received by *RSU1*.

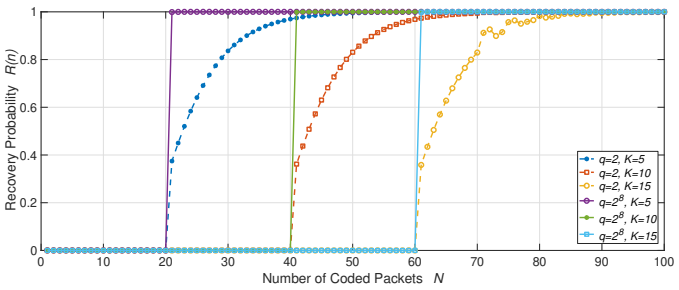

Fig. 7. The recovery probability R as a function of the number of RLNC-CAM transmissions, for $K = 5, 10, 15$ and $q = 2, 2^8$ as received by *RSU2*.

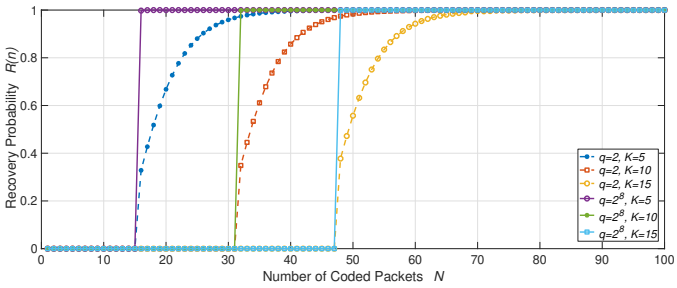

Fig. 8. The recovery probability R as a function of the number of RLNC-CAM transmissions, for $K = 5, 10, 15$ and $q = 2, 2^8$ as received by *RSU3*.

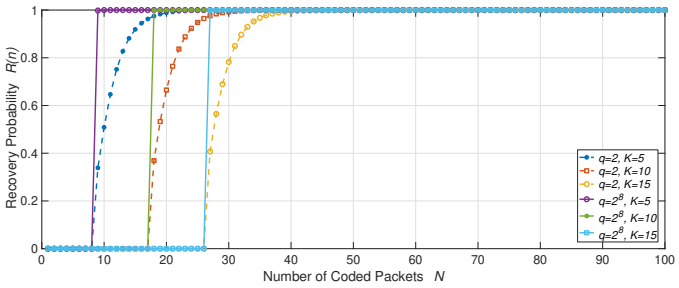

Fig. 9. The recovery probability R as a function of the number of RLNC-CAM transmissions, for $K = 5, 10, 15$ and $q = 2, 2^8$ as received by *RSU4*.

Similarly, Fig. 5 shows the RLNC-CAM Packet Error Rate (PER) for *RSU1* as a function of the distance. We see that when a CAV is in close proximity to an RSU ($\leq 35\,\mathrm{m}$), the PER is $\simeq$20%. For distances between $35\,\mathrm{m}$ to $160\,\mathrm{m}$, the PER fluctuates around $\simeq$40%. Finally, for greater distances, the PER values increase at the range of $75\% - 90\%$. As this figure is based on real-world data, we can observe that the PER does not increase smoothly, but it fluctuates. This is similar to what we observed in [17] and proves the necessity for a Fog Computing Infrastructure and the integration for RLNC principle in the ITS-G5 stack. The PER for the rest of the RSUs behaves similarly. More specifically, for *RSU4* the performance is slightly better in terms of PER and compared to *RSU1*, for *RSU2* is slightly worse, while for *RSU3* is almost identical. The different plots for the rest of the RSUs will not be presented in this work, due to the limited space.

Figs. 6–9 show the recovery probability R as a function of the number of transmitted coded packets. R was individually evaluated for the four RSUs deployed in order to investigate the effect the surrounding environment can introduce. Different source message lengths K were evaluated, namely, $K \in \{5, 10, 15\}$, as well as different sets of N coded packets, with $N \in \{K, \ldots, 100\}$. Finally, two different lengths for the finite field coefficients $g_{i,j}$ where considered, being $q \in \{2, 2^8\}$. In order to validate our solution by testing multiple combinations of RLNC parameters, R was investigated by means of a Monte Carlo simulation, based on the real-world traces acquired from our field trials. In particular, by employing our experimental PER, we calibrated our full-stack network simulator as per [18] and recreated the network scenario shown in Fig. 4.

Fig. 6, for both values of q, shows that, for $K = 5, 10$ and 15, at least $17, 34$ and 50 packets are required to have a chance of recovering a RLNC-CAM, respectively. The values for the rest of the RSUs are $21, 41, 61$ for *RSU2*, $16, 32, 48$ for *RSU3*, and finally, $9, 18, 27$ for *RSU4*. The difference between the four RSUs arises from the different positions and environments where they are located. As we commented before, *RSU4* achieved the best PER performance, being the

RSU positioned on a wide road. As shown, this is reflected in the number of packets required for the recovery of a source message, as well.

In general, as expected, for $q = 2$, we observe that the number of RLNC-CAM transmissions needed to recover a source message is larger than or equal to the case when $q = 2^8$. As noted in [12], larger values of q increase the computational complexity of the RLNC decoder. This is generally a problem in mobile devices with limited computational resources. In our case, the aforementioned issue does not apply as a cloud-based service is responsible for performing the RLNC decoding operations – thus, it is reasonable to assume to have high performance computing capabilities at our disposal. For these reasons, our results lead us to recommend a value of q equal to 2^8.

V. Conclusions

This paper addressed the pressing concern of providing a city-scale Fog Computing architecture enabling the offloading of vehicular sensor data. We propose a Fog Computing infrastructure interconnecting a number of RSUs that in turn, collected sensor data transmitted by CAVs. Both RSUs and CAVs communicate by means of the ETSI's ITS-G5 communication stack that we extended by integrating an RLNC-Facility Sublayer into the ITS-G5 Facility layer. We also propose CAVs to offload their sensor data in a network coded fashion employing a novel type of CAM (namely, the RLNC-CAM). Building upon channel conditions measured during large-scale car trials and utilizing Monte Carlo simulations, we investigated both the feasibility and effectiveness of our proposal.

Acknowledgment

This work is part of the FLOURISH Project, which is supported by Innovate UK, under Grant 102582.

References

[1] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1617–1655, Sep. 2016.

[2] K. Zheng, Q. Zheng, H. Yang, L. Zhao, L. Hou, and P. Chatzimisios, "Reliable and Efficient Autonomous Driving: The Need for Heterogeneous Vehicular Networks," *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 72–79, Dec. 2015.

[3] A. Tassi, M. Egan, R. J. Piechocki, and A. Nix, "Modeling and design of millimeter-wave networks for highway vehicular communication," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 10 676–10 691, Dec. 2017.

[4] M. Levi, Y. Allouche, and A. Kontorovich, "Advanced Analytics for Connected Car Cybersecurity," in *Proc. of IEEE VTC-Spring 2018*, June 2018, pp. 1–7.

[5] M. Sookhak, F. R. Yu, Y. He, H. Talebian, N. S. Safa, N. Zhao, M. K. Khan, and N. Kumar, "Fog Vehicular Computing: Augmentation of Fog Computing Using Vehicular Cloud Computing," *IEEE Veh. Technol. Mag.*, vol. 12, no. 3, pp. 55–64, Sep. 2017.

[6] C. Huang, R. Lu, and K. K. R. Choo, "Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 105–111, Nov. 2017.

[7] M. Riveiro, M. Lebram, and M. Elmer, "Anomaly Detection for Road Traffic: A Visual Analytics Framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2260–2270, Aug 2017.

[8] H. Niu, M. Iwai, K. Sezaki, L. Sun, and Q. Du, "Exploiting Fountain Codes for Secure Wireless Delivery," *IEEE Commun. Lett.*, vol. 18, no. 5, pp. 777–780, May 2014.

[9] G. G. M. N. Ali, M. Noor-A-Rahim, P. H. J. Chong, and Y. L. Guan, "Analysis and Improvement of Reliability Through Coding for Safety Message Broadcasting in Urban Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6774–6787, Aug 2018.

[10] "EN 302 663," ETSI, Tech. Rep., 2013.

[11] E. Magli, M. Wang, P. Frossard, and A. Markopoulou, "Network Coding Meets Multimedia: A Review," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1195–1212, Aug 2013.

[12] A. Tassi, I. Chatzigeorgiou, and D. E. Lucani, "Analysis and Optimization of Sparse Random Linear Network Coding for Reliable Multicast Services," *IEEE Trans. Commun.*, vol. 64, no. 1, Jan. 2016.

[13] A. Festag, "Cooperative Intelligent Transport Systems Standards in Europe," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 166–172, Dec. 2014.

[14] "EN 302 637-2," ETSI, Tech. Rep., 2014.

[15] T. Mladenov, S. Nooshabadi, and K. Kim, "Efficient GF(256) raptor code decoding for multimedia broadcast/multicast services and consumer terminals," *IEEE Trans. Consum. Electron.*, vol. 58, no. 2, pp. 356–363, May 2012.

[16] C. Khirallah, D. Vukobratovic, and J. Thompson, "Performance Analysis and Energy Efficiency of Random Network Coding in LTE-Advanced," *IEEE Trans. Wireless Commun.*, vol. 11, no. 12, pp. 4275–4285, Dec. 2012.

[17] I. Mavromatis, A. Tassi, R. J. Piechocki, and A. Nix, "A City-Scale ITS-G5 Network for Next-Generation Intelligent Transportation Systems: Design Insights and Challenges," in *Ad-hoc, Mobile, and Wireless Networks*, N. Montavont and G. Z. Papadopoulos, Eds. Cham: Springer International Publishing, 2018, pp. 53–63.

[18] ——, "Agile Calibration Process of Full-Stack Simulation Frameworks for V2X Communications," in *Proc. of IEEE VNC 2017*, Nov 2017, pp. 89–96.